International Journal of Security and Its Applications
Vol. 11, No. 12 (2017), pp.15-26
http://dx.doi.org/10.14257/ijsia.2017.11.12.02

Enhancement of Cloud Security using DNA Inspired Multifold Cryptographic Technique

Manreet Sohal[1*] and Sandeep Sharma[2]

[1]*Department of Computer Engineering and Technology, Guru Nanak Dev University, Amritsar-143005, India*
[2]*Department of Computer Engineering and Technology, Guru Nanak Dev University, Amritsar-143005, India*
[1]*manreet.cetrsh@gndu.ac.in and* [2]*sandeep.cse@gndu.ac.in*

Abstract

One of the major hurdles, in the way of cloud computing is data security. To triumph over this, many security mechanisms have been adopted. Cryptography is very beneficial among them because albeit the data can be accessed without authorization, it cannot be read. The Data owners must encrypt their data prior storing it on the cloud. This guarantees the secrecy of the crucial data which is accumulated on the public cloud storage. There are lots of cryptographic techniques that have been developed to ensure confidentiality but, most of them are based on complex mathematical calculations and equations. In this paper, a novel and unique encryption algorithm have been proposed which is inspired by DNA cryptography. The proposed algorithm is highly secure as it makes use of random encoding tables and it is very hard to speculate the original sequence. It provides multifold security. Certain transformations are applied over the encryption key before using it. Here, each character is encoded as a 7 bit character and further, as 5 bit character. The encoding is highly robust; therefore it is very difficult to break the ciphertext. The encryption process is highly dynamic. The same plaintext yields different ciphertext for every session.

Keywords*: Cloud Computing, Security, Cryptography, Cloud Data Encryption, Encoding*

1. Introduction

By making use of internet technologies, cloud computing has made a great contribution in the industries. But, in addition to that, cloud computing has also been adopted at an individual level by people due to its enormous benefits. Capital disbursements have been trimmed down significantly and the suppleness of carrying out the business processes has altered the industrial scenario substantially. Various benefits like IaaS (Infrastructure as Services), PaaS (Platform as Services), SaaS (Software as services), disaster recovery *etc.*, have been provided by the cloud [1]. The perception of cloud computing was used in reality for the first time by the Amazon Web Services. During the time of no max out usage of IT resources, the company leased its resources to other companies on demand basis [2].

There is no doubt about the espousement of cloud computing. Therefore, the stress should be laid on the point when one should start using cloud computing. Business services facilitated by IT sectors are being distributed by cloud computing in huge numbers. Virtualized computing environments, which can be scaled vigorously, can be attained by implementing cloud computing with reasonable

Received (September 5, 2017), Review Result (November 17, 2017), Accepted (November 22, 2017)
* Corresponding Author

ISSN: 1738-9976 IJSIA

resources and costs. By making use of cloud technologies, the overrated IT resources like hardware, infrastructure, software, expertise *etc* can be used on pay per demand basis [1]. The resources can be used by the organization on demand without the need of managing them. The cloud providers are responsible for the maintenance of cloud environment.

Lots of people use services of the cloud without being aware of it. For example, while using any version of email (yahoomail, gmail) or while using any of the applications that have not been physically mounted on the local pc like MS Word, Excel *etc* [2], the customers might not be aware of the location of the servers that store their emails and unaware of the servers that host the source codes of applications used by them. The cloud services come from the datacenters by making use of the concept of virtualization. But, the widespread acceptance and implementation of cloud computing is only beneficial, if the security is guaranteed. There are two major questions that always confronts cloud computing. 1. How to assure the best security of data [1]? 2. How to keep the private information of the clients confidential [1]? In the list of cloud security risks, data protection is at the top.

1.1. Security Challenges in Cloud

Cloud and virtualization provides us with the nimbleness and effectiveness of instantaneously rolling out the new services. But, the deficiency of an appropriate physical control, fetches an entire horde of data security issues, such as co-mingling of data, data leakage and deletion, misuse by privileged users, backups, data replications, lack of visibility *etc*. Such issues lead to the terrific anxiety, concerning the security hazards in the cloud. Organizations are anxious about trusting their employees or they have to employ supplementary in-house controls in the private cloud, and are not sure that the third-party service providers can offer ample security in multitenant environments in which competitor's data may also be stored [3]. There is also constant apprehension about the protection of data, which is moving between the organization and the cloud, plus how to make it certain that there is no left over data remnants while migrating to another cloud service provider [3].

Indubitably, the virtualized environments and the private cloud embrace new dares in the safety of data, mixed trust levels, and the probable abating of division of duties and data governance [3]. The public cloud mixes these threats with data which is easily moveable, can be accessed by anyone who connects with the cloud server, and can be duplicated for ease of use. The challenge faced by hybrid clouds is to secure data as it travels between an enterprise and a public cloud. There are specific security challenges associated with each of the three cloud service models—Software as a Service (SaaS), Platform as a Service (PaaS), and Infrastructure as a Service (IaaS) [3].

- **SaaS** allows the customers to utilize the applications of the cloud service provider, running on cloud infrastructure through the internet. But, besides presenting anywhere access, it also raises up the security threats. With this model, it has become important to execute identity management and access control policies. For example, in salesforce.com quite few salespersons are allowed to download and access private sales information of the customers [3].
- **PaaS** provides a framework, where the applications deployed by the customers can be managed and executed, but the underlying cloud infrastructure is not under their control [3]. In this type of model, a strong authentication is required to indentify the users. Besides this, an audit trail and a compliance regulatory support are also needed.

- **IaaS** provides a hardware infrastructure as virtualized systems that can be accessed via internet. Network, storage, memory, processor and several other computing resources are provided under this model. The underlying resources are under the control of cloud service provider. One such service is provided by Amazon's Elastic Compute Cloud (EC2). Security of data is a collaborative responsibility of the cloud service provider and the cloud user. Encrypting the data without modifying applications has become the key necessity of the IaaS environment [3].

The rest of the paper is organized as follows. Section 2 discusses about the cloud data encryption. In section 3 various cloud encryption techniques from the literature have been discussed. Section 4 highlights the requirements for the strong encryption technique. Section 5 discusses about the proposed technique. Section 6 provides the conclusions of this paper.

2. Cloud Data Encryption

Besides various benefits, cloud computing leads to the loss of control. Once the data is housed on the third party servers, there is no surety that it is secure. Although, the cloud service providers assure it with their promises of cloud encryption, it is not certain that they would not hand over one's data to the government agencies without one's knowledge or approval. In addition to it, as a part of their daily operations, the cloud providers make copies of user's data, move it and perform its backup. The cloud data encryption resolves many of the control issues that the enterprises come across while using the cloud. Even though the cloud service providers are forced to reveal the data, but the data which has been encrypted cannot be read by the unauthorized users, provided that the enterprises retain control of their encryption keys. Moreover, placing the focus on the data rather than on infrastructure makes it certain that data will remain secure even though there are hardware vulnerabilities.

Cloud encryption is converting the data of cloud customers into ciphertext [4]. The only difference between cloud encryption and in house encryption is that cloud encryption requires the customer to gain knowledge of the policies and encryption methods used by the service provider. The security provided by the cloud service providers must match with the level of sensitivity of the data being hosted. Encryption is not a new technology but, traditionally the data which was encrypted was kept on the servers residing on premises, which were under the direct control of the companies. Since, today most of the in-style business applications are hosted in the cloud, business management is either required to depend upon the contract language to safeguard their assets, by choosing a appropriate cloud provider that will permit the customer to encrypt the data prior sending it to the cloud for storage or processing, or establish a partnership with software as a service (SaaS) provider that will deal with the encryption and decryption of the commercial data. Once the data arrives at the servers of the cloud service providers, the application provider normally encrypts it to protect the data at rest [5].

The cloud encryption can be provided at both the provider side and at the client side. The former approach is followed by the leading cloud service providers like Microsoft, Google, and Yahoo. The enterprises which have taken up any of the popular public cloud services like Salesforce, Dropbox, or Google adopt client side data encryption. There are many cloud encryption gateways that allow the enterprises to identify and encrypt sensitive data before it leaves the enterprise premises. But, for these methods, the enterprises need to invest in the infrastructure, but these can free them from the promises of unwillingly trusted cloud service

providers. The enterprises have full control of their data and even the service providers cannot access that data.

In the next section various cloud encryption techniques that already exist in the literature have been discussed. The major claims of these techniques have been highlighted.

3. Existing Work

In 2013, Z.Jia *et al*. [6] presented a new technique based on eCryptfs (Enterprise File Encryption system), which is in combination with the whole architecture of private clouds. In this paper, the authors have analyzed the benefits of securing private cloud in the enterprise. The authors have claimed that with the proposed technique amalgamation of internal resources can be recognized, resource utilization rate can be improved and the operating costs can be reduced.

In 2015, P.Ora *et al*. [7] have proposed a new technique for maintaining security and integrity of data on the cloud servers. The authors have used partial homomorphic algorithm based on RSA for encryption and decryption and for securely backing up the data MD5 hashing technique has been used. The proposed technique involves 4 steps: Encryption/Decryption, uploading data on the cloud, generation of hash values and data verifications.

In 2015, N.Shimbre *et al*. [8] have discussed about data security problems in distributed data storage systems. The authors have discussed about security concerns like fast localization of errors and security and integrity of data. In this paper, techniques for generating hash codes and third party auditing has been presented. The authors have used AES algorithm and SHA-1.

In 2015, S.Kumar *et al*. [9] have analyzed various attribute based encryption algorithms and have discussed the merits and demerits of these approaches. Further, the authors have proposed a new method of encryption by making use of digital signature, asymmetric approach of encryption and hash functions. This proposed approach is simplified version of attribute based encryption.

In 2015, S.Huang *et al*. [10] have proposed a new alternative of predicate encryption. This model has the properties of data self destruction and timed released services. The authors have further provided the extended version of their proposed model which provides additional features of encryption of long messages and undecryptable search. It has been claimed that the proposed model is appropriate for encrypted data search on cloud.

In 2015, Y. Lu *et al*. [11] proposed a new method of certificate based proxy re-encryption that does not use bilinear pairings. It has been proved that this system is more secure under the assumption taken in random oracle model using computational Diffie Hellman approach. Since, the proposed model does not use bilinear pairing; the authors have claimed that its computational cost is low compared to other models of certificate based proxy re-encryption that uses bilinear pairing.

In 2015, J. Li *et al*. [12] have considered the problems of authorized data Deduplication. In contrast to the existing Deduplication systems, the authors have further taken into account the duplicate checks alongside the data itself. In this paper, new Deduplication approaches have also been proposed that are applied in hybrid clouds.

In 2016, S. Ma [13] has presented a new tool for cryptography known as an Identity based Encryption with equality test. In this paper, in opposition to a selected identity attack, a one-way chosen cipher text security has been defined. The authors have given the construction of their model using bilinear pairing.

In 2016, E.Hossain *et al*. [14] have presented a technique based on DNA cryptography. In this paper, dynamic sequence tables and concept of OTP has been used to enhance the security levels. This is a very strong algorithm because to break it, the attacker needs to perform all the possible combinations which is next to impossible. The technique is very secure against brute force attacks.

Almost all of these algorithms are based on conventional cryptographic systems which have an extensive bequest. All of these techniques involve a strong mathematical and theoretical foundation. But, still conventional security systems like RSA, DES, and AES are also being implemented in actual time operations. So, there is a need to develop a perception that the newly proposed technique does not negate the custom, instead generate a bridge between the existing and the new technology. The technique proposed in this paper follows the idea of DNA cryptography. This approach will strengthen the existing security systems by opening up a new possibility of a hybrid cryptographic system. In this paper, we are proposing a cryptographic technique in which the user encrypts his/her data before sending it to the cloud *i.e* it uses client side data encryption. Every time any authorized user wants to access the data, a 14 bit key is sent to the user along with encrypted data. By using this key, the user can decrypt the data. All the authorized users are provided with the encoding tables, beforehand. The keys are managed by the cloud service provider.

4. Prerequisites for a Strong Encryption Algorithm

For providing high security**,** every encryption algorithm must fulfill certain set of requirements. Based upon the existing literature, the following requirements have been identified that should be satisfied by the newly proposed algorithm:

- Encoding of the complete set of characters
- Encoding of each character of the plaintext into unique sequence
- Encoding should be robust
- Encryption process should be dynamic

5. The Proposed Technique

The methodology proposed in this paper, is a novel cryptographic technique to enhance cloud security. The general idea of the proposed technique is somewhat based upon the idea of DNA cryptography presented in [14], but it does not use DNA cryptography as it is practically very difficult to implement it. In this paper, a more random and hence, a more secure technique has been designed. In this technique, firstly a random table is encoded which contains all the possible combinations that can be formed out of 7 bits. Therefore, the table contains 128 different bit combinations which can be used to represent 96 most widely used ASCII characters. Our table can be expanded to represent 32 more characters if required. After creating 128 different 7-bit patterns, each of the 96 ASCII characters are assigned to these 7-bit patterns on purely random basis. Each character gets a unique 7-bit sequence irrespective of its position in the ASCII table and irrespective of the ASCII value assigned to it. The assignment is completely random which makes this encoding stronger. Generally, each character is represented using 8 bits.

In the existing systems, even if the bits have been scrambled or have been subjected to any transformation, it is somewhat predictable that a character would be represented by 8 bits. But, in the proposed approach the bits for one character have been reduced to seven. Even in DNA cryptography, each character is represented as the combination of four DNA bases *i.e* A, T, C, G which form the DNA sequences. Further, each base is represented by 2 bits, for example A=00,

T=01, C=10 and G=11 [14]. Thus, the combinations of these four bases also represent each character by 8 bits.

Secondly in [14], the key is of the same length as that of the data. Though, it would be very secure but, it almost doubles the amount of data that has been sent to the receiver end. In the approach presented in this paper, initially a 14 bit key is assumed which is generated randomly. Then, a 7 bit key is extracted from it by checking whether the first bit of the key is 0 or 1. This significantly reduces the size of extra bits sent along with the original data. Thus, the transmission and storage costs are reduced. It provides multifold security. Even if someone is able to gain access to the key, it would be mistaken as if the whole key has been used for encryption.

The third improvement is over Table 2. In [14], before sending the DNA sequence to the receiver, it is finally converted into amino acids using Table 2. In this Table 26 alphabets from A to Z were used to represent different amino acids. Here, each alphabet from A-Z represents multiple amino acids sequences. For example, assume that A represents or encodes three different amino acid sequences. In this case, the problem arises during decryption. Now for each occurrence of A in the ciphertext it would not be clear which amino acid sequence should be replaced with A. This leads to ambiguity. Moreover, it uses only upper case letters. Therefore, the ciphertext would contain only upper case letters. The attackers become suspicious that the original data has been encoded in these upper case letters.

In technique designed in this paper, Table 2 is constructed by considering all possible combinations of 5 bits *i.e* 32 combinations. To each of this 5-bit combination a unique character from Table 1 is assigned randomly. We have selected 32 random characters from Table 1. In this way Table 2 will contain different characters and cover more categories like uppercase letters, lower cases letters, numbers and special symbols. Therefore, the ciphertext will contain all these categories and it would be make it more random and secure. After encoding the original plaintext into 7-bit sequences using Table1, the bit sequence goes through series of steps performing various operations like substitution and transposition before converting them into 5 bit characters using Table 2. The output obtained by using Table 2 is then sent to the receiver. It is assumed that the two encoding tables are present at both the sender and receiver site.

The proposed algorithm consists of two parts:

- Generation of encoding table
- Encryption and decryption algorithms

5.1. Generation of Encoding Tables

The generation of the two encoding tables is purely random. First of all, all the combinations that can be made using 7 bits are generated for encoding Table 1. This results in 128 different bit patterns. After this, 96 ASCII characters are assigned to these bits randomly irrespective of their ASCII values. Only 96 bit patterns are assigned to the ASCII characters, 32 random bit patterns are left unassigned which further add up to the randomness of the table. This table is generated after every session intervals which makes the decoding even more difficult.

For the encoding of Table 2, all combinations of 5 bits are generated and 32 random characters from Table 1 are assigned to these 5-bit sequences. The assigned characters have all the categories of characters like uppercase, lowercases, numbers and special characters.

Table 1. Encoding of ASCII Characters as 7-bit Combinations

0000000-y	0000001-\	0000010-l	0000011-,	0000100-;	0000101
0000110-W	0000111-a	0001000-J	0001001-$	0001010-‘	0001011
0001100-{	0001101-g	0001110-7	0001111-r	0010000-w	0010001
0010010-b	0010011-E	0010100-C	0010101-P	0010110-U	0010111
0011000-z	0011001-2	0011010-m	0011011-.	0011100-”	0011101
0011110-X	0011111-+	0100000-K	0100001-#	0100010-space	0100011
0100100-[	0100101-h	0100110-8	0100111-&	0101000-x	0101001
0101010-c	0101011-F	0101100-*	0101101-s	0101110-V	0101111
0110000-A	0110001-3	0110010-n	0110011-Q	0110100-,	0110101
0110110-Y	0110111-=	0111000-L	0111001-?	0111010-`	0111011
0111100-}	0111101-i	0111110-9	0111111-@	1000000	1000001
1000010-d	1000011-G	1000100-o	1000101-t	1000110	1000111
1001000-B	1001001-4	1001010-M	1001011-R	1001100	1001101
1001110-Z	1001111-_	1010000-<	1010001-/	1010010	1010011
1010100-]	1010101-j	1010110-^	1010111-!	1011000	1011001
1011010-e	1011011-H	1011100-p	1011101-u	1011110	1011111
1100000-(	1100001-5	1100010-N	1100011-S	1100100	1100101
1100110-0	1100111--	1101000->	1101001-:	1101010	1101011
1101100-|	1101101-k	1101110-%	1101111-~	1110000	1110001
1110010-f	1110011-I	1110100-q	1110101-v	1110110	1110111
1111000-D	1111001-6	1111010-O	1111011-T	1111100	1111101
1111110-1	1111111-)				

Table 2. Final Encoding Table

00000-T	01000-s	10000-l	11000-1
00001-A	01001-G	10001-f	11001-,
00010-b	01010-Q	10010-J	11010-o
00011-;	01011-p	10011-m	11011-5
00100-C	01100-h	10100-Y	11100-.
00101-X	01101-i	10101-$	11101-w
00110-v	01110-r	10110-k	11110-3
00111-N	01111-D	10111-Z	11111-space

5.2. Encryption and Decryption Algorithms

The following section involves the discussion of the proposed methods of encryption and decryption.

5.2.1. Encryption Process

The encryption process for encrypting plaintext into ciphertext consists of the following steps:

Let us assume that the plaintext is “Stay Happy”

Step 1: Convert the given string into 7 bit sequence using Table 1.

Step 2: Generate a random 14 bit key.

11001101001101

If the first Digit of the key is 1, extract all the digits at odd positions.

Else extract all the digits at even positions.

Therefore the resulting 7 bit key is 1010010

Step 3: Perform the XoR operation between the 7-bit sequence of the plaintext and the extracted key.

Plaintext:

1100011100010100001110000000010001010110110000111101110010111000000000

Key:

1010010101001010100101010010101001010100101010010101001010100101010010

The answer of XoR is
01100010010111101010110100101110000000100110101010001110000111010100 10

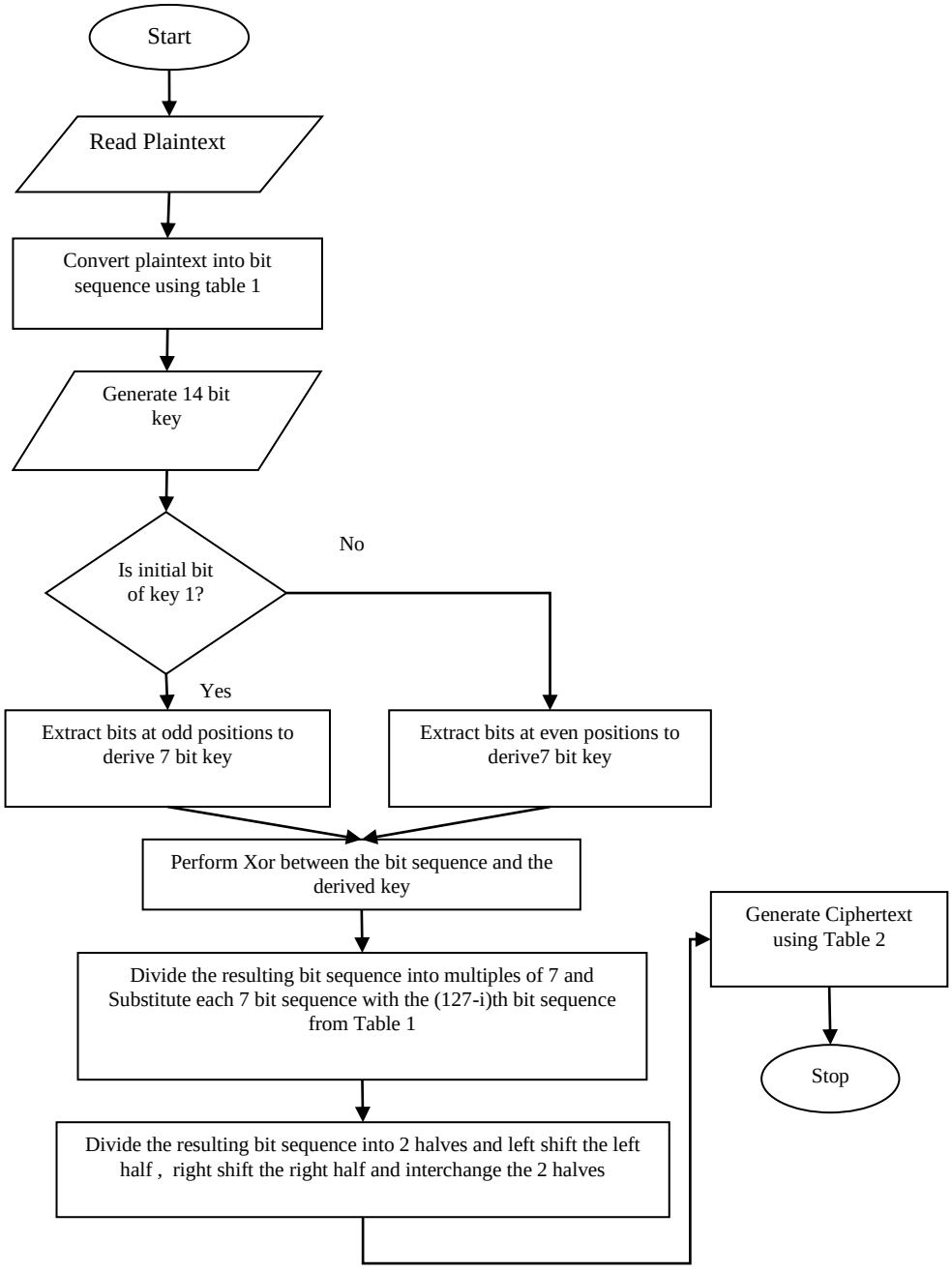

Figure 1. The Encryption Process

Step 4: Now perform substitution on the bits sequence obtained after step 3. Divide the bits sequence into multiples of 7 and substitute each bit sequence (say i) with (127-i)th bit sequence using Table 1.

1000110 1100110 0100010 0100101 0000111 1101000 0100010 1100011 1100011 0100101

Step 5: This step involves transposition:
a. Divide the bit sequence into two halves
10001101100110010001001001010000111 & 11010000100010110001111000110100101
b. Left shift the left half
00011011001100100010010010100001111
c. Right shift the right half
11101000010001011000111100011010010
d. Interchange the two halves
11101000010001011000111100011010010000110110011001000100100101000011111

Step 6: Now divide the bit sequence into 5-bit sequences and pad 0'z at the end if necessary.
11101 00001 00010 11000 11110 00110 10010 00011 01100 11001 00010 01001 01000 01111

Now generate ciphertext using Table 2.
The ciphertext generated is : wAb13vJ;h,bGsD

5.2.2. Decryption Process

The decryption process is the reverse of encryption process. It involves the following steps for converting the ciphertext into plaintext.

Step 1: Change the ciphertext into bit sequence using Table 2.
Ciphertext: wAb13vJ;h, bGsD
The Bit sequence obtained is:
11101000010001011000111100011010010000110110011001000100100101000 01111

Step 2: Apply reverse transposition.

a. Divide the bit sequence into 2 halves.
11101000010001011000111100011010010 & 00011011001100100010010010100001111

b. Interchange the two halves.
00011011001100100010010010100001111 & 11101000010001011000111100011010010

c. Right shift the first half.
10001101100110010001001001010000111

d. Left shift the second half.
11010000100010110001111000110100101

e. Merge the two halves.
10001101100110010001001001010000111110100001000101100011110001101001010101

Step 3: Now perform reverse substitution on the bits sequence obtained after step 2. Divide the bit sequence into multiples of 7 and substitute each bit sequence (say i) with (127-i)th and vice versa bit sequence using Table 1.

0110001 0010111 1010101 1010010 1110000 0001001 1010101 0001110 0001110 1010010

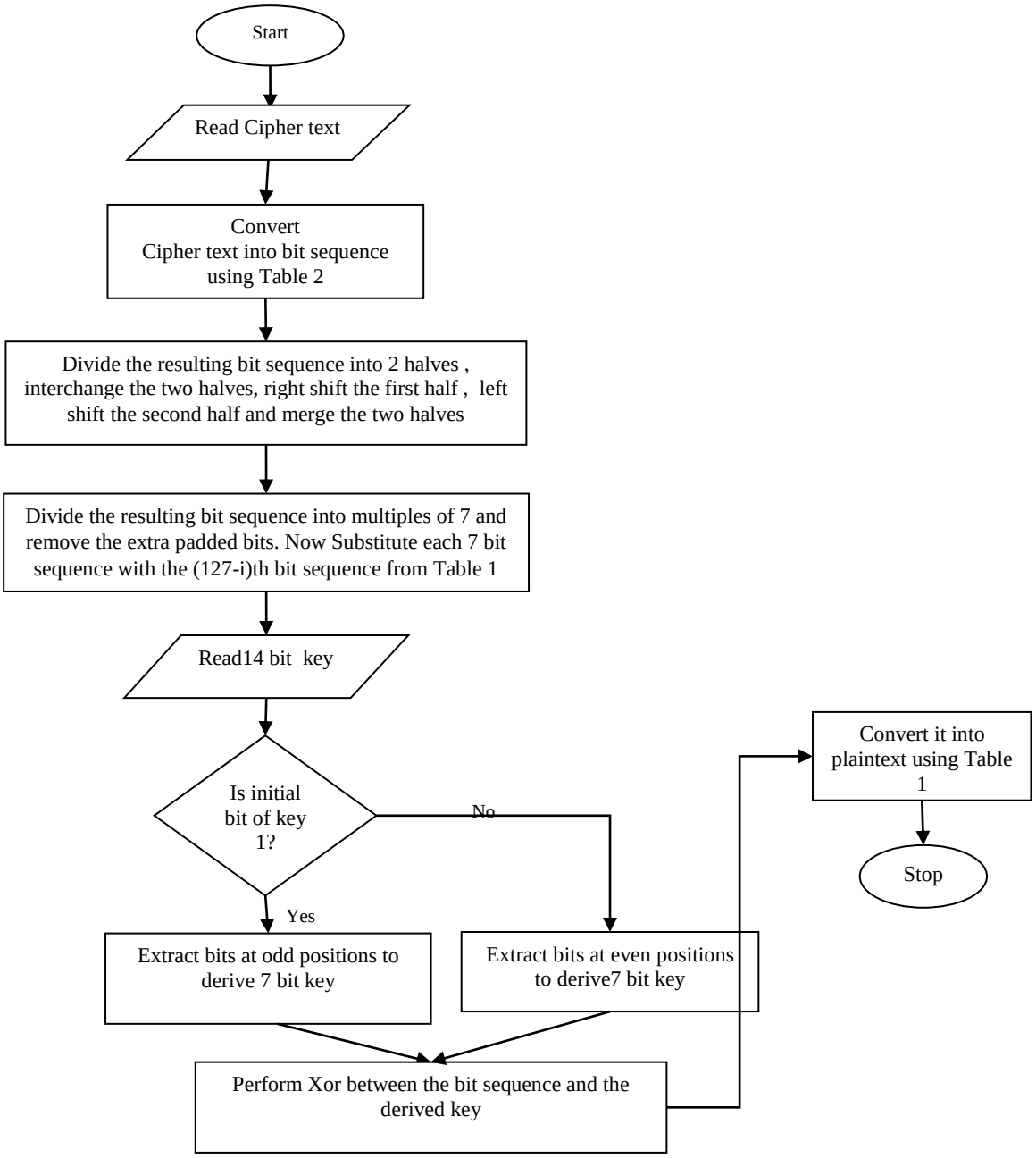

Figure 2. The Decryption Process

Step 4: Extract 7 bit key from the 14 bit key (11001101001101) sent by the sender according to following rule:

If the first Digit of the key is 1, extract all the digits at odd positions.

Else extract all the digits at even positions.

Therefore the resulting 7 bit key is 1010010

Step 5: Perform the XoR operation between the 7-bit sequence obtained in step 3 and the extracted key.

Bit sequence from step3:

01100010010111101010110100101110000000100110101010001110000111
01010010

Key:
10100101010010101001010100101010010101001010100101010010101001010100
10

The resulting plaintext in bit sequence:1100011 1000101 0000111 0000000 0100010 1011011 0000111 1011100 1011100 0000000

Step 6: By using Table 1 convert each 7-bit sequence into the corresponding alphabet.

1100011 1000101 0000111 0000000 0100010 1011011 0000111 1011100 1011100 0000000

S t a y H a p p y

5.3. Fulfillment of the Desired Requirements

- **Encoding of the complete set of characters:** This requirement is met as each of the 96 ASCII character has been encoded in Table 1.
- **Encoding of each character of the plaintext into unique sequence:** From Table 1, it is clear that each character is assigned to a unique 7 bit pattern
- **Encoding should be robust:** It has been shown that the encoding table is generated on completely random basis. Each character has been assigned the bit sequences irrespective of their positions ad values in the ASCII table.
- **Encryption process should be dynamic:** This condition is met as the same plaintext yields different ciphertext for every session as encoding table is regenerated after each session. This makes the encryption process more secure and dynamic.

6. Conclusions

In this paper, a novel cryptographic technique has been proposed. Unlike the existing techniques, it does not involve complex mathematical equations and formulas. The proposed technique is inspired by the way, the DNA cryptography works and the main idea has been taken from it. Though, the DNA cryptography is a very strong encryption technique, but it has a major drawback that it is practically very difficult to implement. It is quite difficult to synthesize various DNA strands to hide data in it. Due to this, it is not becoming mature in theory as well as in realization. Therefore, our framework is the binary form of DNA cryptography as it works in a similar manner, but can be implemented very easily as it does not involve the actual biological processes. This proposed technique removes the flaws which where discerned in the existing DNA algorithm, hence making the cryptographic algorithm stronger. The proposed algorithm use 7 bits to represent a character unlike the convention of denoting each character with a byte. Further, the character is converted into 5 bit sequence. This new technique provides multifold security. The encryption key is further transformed before actually using it for encryption. The encoding tables are purely random and are reconstructed after each session. This makes the encryption more random and dynamic. This will give in a different ciphertext for the same plaintext in different sessions. Thus, the algorithm proposed in this paper is highly secure and robust.

References

[1] M. Tebaa and S.E. Hajji, "Secure Cloud Computing Through Homomorphic Encryption", International Journal of Advancements in Computing Technology, vol. 5, no. 16, **(2014)**.
[2] N. Sengupta and R. Chinnasamy, "Contriving Hybrid DESCAST Algorithm for Cloud Security", Procedia Computer Science, vol. 54, **(2015)**, pp. 47-56.
[3] Data Security in the Cloud. [online] Vormetric Data Secuirty Simplified, Custom Solution Groups. http://enterprise-encryption.vormetric.com/rs/vormetric/images/wp-cso-vormetric-data-security-in-the-cloud-updated.pdf. PDF File.
[4] M. Rouse, "Cloud Encryption (Cloud Storage Encryption)", [Online] searchcloudstorage.techtarget.com. http://searchcloudstorage.techtarget.com/definition/cloud-storage-encryption (accessed June 23, 2017).
[5] S. Lawten, "Cloud Encryption: Using Data Encryption in the Cloud", [Online] tomsitpro.com. http://www.tomsitpro.com/articles/cloud-data-encryption,2-913.html (accessed June 23,2017).
[6] Z.P. Jia and X. Tian, "A Novel Security Private Cloud Solution Based on eCryptfs", In IEEE 6th International Conference on Information Management, Innovation Management and Industrial Engineering, Xi'an, China, **(2013)**, pp. 38-41.
[7] P. Ora and P.R. Pal, "Data Security and Integrity in Cloud Computing Based on RSA Partial Homomorphic and MD5 Cryptography", In IEEE International Conference on Computer, Communication and Control, Indore, India, **(2015)**, pp. 1-6.
[8] N. Shimbre and P. Deshpande, "Enhancing Distributed Data Storage Security for Cloud Computing Using TPA and AES Algorithm", In IEEE International Conference on Computing Communication Control and Automation, Pune, India, **(2015)**, pp.35-39.
[9] N.S. Kumar, G.V.R. Lakshmi and B. Balamurugan, "Enhanced Attribute Based Encryption for Cloud Computing", Procedia Computer Science, vol. 46, **(2015)**, pp. 689-696.
[10] S.Y. Huang, C.I. Fan and Y.F. Tseng, "Enabled/Disabled Predicate Encryption in Clouds", Future Generation Computer Systems, vol. 68, **(2016)**, pp. 148-160.
[11] Y. Lu and J. Li, "A Pairing-free Certificate-based Proxy Re-Encryption Scheme for Secure Data Sharing in Public Clouds", Future Generation Computer Systems, vol. 62, **(2016)**, pp. 140-147.
[12] J. Li, Y.K. Li, X. Chen, P.P. Lee and W. Lou, "A Hybrid Cloud Approach for Secure Authorized Deduplication", In IEEE Transactions on Parallel and Distributed Systems, vol. 26, no. 5, **(2015)**, pp. 1206-1216.
[13] S. Ma, "Identity-based Encryption with Outsourced Equality Test in Cloud Computing", Information Sciences, vol. 328, **(2016)**, pp. 389-402.
[14] E.M.S. Hossain, K.M.R. Alam, M.R. Biswas and Y. Morimoto, "A DNA Cryptographic Technique Based on Dynamic DNA Sequence Table", In IEEE 19th International Conference on, Computer and Information Technology (ICCIT), Dhaka, Bangladesh, **(2016)**, pp. 270-275.

Authors

Manreet Sohal, she is a research scholar, pursuing PhD at Department of Computer Engineering & Technology, Guru Nanak Dev University Amritsar. She has passed out her M.Tech (CSE) and B.Tech (CSE) from Guru Nanak Dev University. Her main area of interest is Cloud Computing.

Sandeep Sharma, he is a Professor and The Head of Department of Computer Engineering and Technology at Guru Nanak Dev University, Amritsar. He is a Chief Security Information Officer as well as Nodal Officer of Digital India Week. He is the Network as well as mail Administrator of the Guru Nanak Dev University. He has graduated with B.E (Computer) degree from the University of Pune, received the M.E (Computer) from Thapar University Patiala and received his PhD degree from Guru Nanak Dev University Amritsar. He has many research publications in the areas of parallel processing, wireless sensor networks and Cloud computing.

